PERGAMON

Pattern Recognition 35 (2002) 1771–1784

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Self-adaptive algorithm of impulsive noise reduction in color images

B. Smolka[a,1,2], K.N. Plataniotis[b,*], A. Chydzinski[a], M. Szczepanski[a,1], A.N. Venetsanopoulos[b], K. Wojciechowski[a]

[a] *Department of Automatic Control, Silesian University of Technology, Akademicka 16 Str, 44-101 Gliwice, Poland*
[b] *Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, 10 King's College Road, Toronto, Canada M5S 1A4*

Received 5 July 2001

Abstract

In this paper a new approach to the problem of impulsive noise reduction in color images is presented. The basic idea behind the new image filtering technique is the maximization of the similarities between pixels in a predefined filtering window. The improvement introduced to this technique lies in the adaptive establishing of parameters of the similarity function and causes that the new filter adapts itself to the fraction of corrupted image pixels. The new method preserves edges, corners and fine image details, is relatively fast and easy to implement. The results show that the proposed method outperforms most of the basic algorithms for the reduction of impulsive noise in color images. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Color image processing; Multichannel filters; Impulsive noise; Adaptive techniques

1. Standard noise reduction filters

A number of non-linear, multichannel filters, which utilize correlation among multivariate vectors using various distance measures have been proposed [1–6]. The most popular nonlinear, multichannel filters are based on the ordering of vectors in a predefined moving window. The output of these filters is defined as the lowest ranked vector according to a specific vector ordering technique.

Let $\mathbf{F}(x)$ represent a multichannel image and let W be a window of finite size n (filter length). The noisy image vectors inside the filtering window W will be denoted as $\mathbf{F}_j$, $j = 0, 1, \ldots, n-1$. If the distance between two vectors $\mathbf{F}_i, \mathbf{F}_j$ is denoted as $\rho(\mathbf{F}_i, \mathbf{F}_j)$ then the scalar quantity $R_i = \sum_{j=0}^{n-1} \rho(\mathbf{F}_i, \mathbf{F}_j)$, is the distance associated with the noisy vector $\mathbf{F}_i$. The ordering of the R_i's: $R_{(0)} \leqslant R_{(1)} \leqslant \cdots \leqslant R_{(n-1)}$, implies the same ordering to the corresponding vectors $\mathbf{F}_i$: $\mathbf{F}_{(0)} \leqslant \mathbf{F}_{(1)} \leqslant \cdots \leqslant \mathbf{F}_{(n-1)}$. Non-linear ranked type multichannel estimators define the vector $\mathbf{F}_{(0)}$ as the filter output. However, the concept of input ordering, initially applied to scalar quantities is not easily extended to multichannel data, since there is no universal way to define ordering in vector spaces.

To overcome this problem, distance functions are often utilized to order vectors. For example, the *vector median filter* (VMF) uses the L_1 or L_2 norm to order vectors according to their relative magnitude differences [2,4,7].

The orientation difference between two vectors can also be used as their distance measure. This so-called *vector angle criterion* is used by the *vector directional*

* Corresponding author. Tel.: +1-416-946-5605; fax: +1-416-978-4425.

E-mail addresses: bsmolka@ia.polsl.gliwice.pl (B. Smolka), kostas@dsp.toronto.edu (K.N. Plataniotis).

[1] Partially supported by KBN Grant 8 T11E 013 19.

[2] Partially supported by NATO Collaborative Linkage Grant LST.CLG.977845.

PII: S0031-3203(01)00169-8

filters (VDF) to remove vectors with atypical directions [5,8].

The *basic vector directional filter* (BVDF) is a ranked-order, nonlinear filter which parallelizes the VMF operation. However, a distance criterion, different from the L_1, L_2 norm used in VMF, is utilized to rank the input vectors. The output of the BVDF is that vector from the input set, which minimizes the sum of the angles with the other vectors. In other words, the BVDF chooses the vector most centrally located without considering the magnitudes of the input vectors.

To improve the efficiency of the directional filters, a new method called *directional-distance filter* (DDF) was proposed [5]. This filter retains the structure of the BVDF but utilizes a new distance criterion to order the vectors inside the processing window.

Another efficient rank-ordered technique called *Hybrid Directional Filter* was presented in Ref. [9]. This filter operates on the direction and the magnitude of the vectors independently and then combines them to produce a unique final output. Another more complex hybrid filter, which involves the utilization of an *arithmetic mean filter* (AMF), has also been proposed [3,9].

All standard filters detect and replace well noisy pixels, but their property of preserving pixels which were not corrupted by the noise process is far from the ideal. In this paper we show the construction of a simple, efficient and fast filter which removes disturbed pixels, but has the ability of preserving original pixel values.

2. Basic algorithm

Let us start from a gray scale image in order to better explain how the new algorithm is constructed. Let the gray scale image be represented by a matrix $\mathbf{F}$ of size $N_1 \times N_2$, $\mathbf{F} = \{F(i,j) \in \{0,\dots,255\},\ i = 1,2,\dots,N_1,\ j = 1,2,\dots,N_2\}$. Our construction starts with the introduction of the similarity function $\mu : [0;\infty) \to \mathbf{R}$. We will need the following assumption for μ:

1. μ is non-ascending in $[0;\infty)$,
2. μ is convex in $[0;\infty)$,
3. $\mu(0) = 1$, $\mu(\infty) = 0$.

As the argument of the function μ will be a distance between pixels in gray scale space, it is easy to understand the sense of 1. The fact that μ must be non-ascending means that the similarity between two pixels is small if the distance between them in a given space is large. Assumption 3 is just a natural normalization of the similarity function.

In this way, the similarity between two pixels with the same gray scale value is 1, the similarity between pixels with far distant intensities is 0. The sense of 2 will be explained below.

In the construction of our filter the central pixel in the window W is replaced by that one, which maximizes the sum of similarities between all its neighbors. Our basic assumption is that a new pixel must be taken from the window W (introducing pixels which do not occur in the image is prohibited like in the VMF and VDF). For this purpose μ must be convex, which is shown by the following Lemma.

Lemma. *Let n numbers $a_1, a_2, \dots, a_n$ be given and the function $f : [0;\infty) \to \mathbf{R}$ be convex. Denote $a = \min\{a_1,\dots,a_n\}$, $b = \max\{a_1,\dots,a_n\}$ and*

$$r(x) = \sum_{i=1}^{n} f(|x - a_i|). \tag{1}$$

Under the above assumptions

$$\max_{x \in [a;b]} r(x) = \max\{r(a_1), r(a_2), \dots, r(a_n)\}. \tag{2}$$

This means that in order to find a maximum of the function $r(x)$ in $[a;b]$ it is sufficient to calculate the values of r only in points $a_1,\dots,a_n$.

Proof of the Lemma. Let us take the longest possible subsequence of m different numbers $a_{(1)} < a_{(2)} < \cdots < a_{(m)}$ from the set $A = \{a_1, a_2, \dots, a_n\}$ and denote by $L_{(i)}$ the number of occurrences of $a_{(i)}$ in A. Clearly

$$r(x) = \sum_{i=1}^{m} L_{(i)} f(|x - a_{(i)}|).$$

Considering function $r(x)$ in one particular interval $[a_{(j)}; a_{(j+1)}]$ gives

$$r(x) = \sum_{i=1}^{j} L_{(i)} f(a_{(i)} - x) + \sum_{i=j+1}^{m} L_{(i)} f(x - a_{(i)}).$$

Obviously functions $f(a_{(i)} - x)$ and $f(x - a_{(i)})$ are convex. In this way, the function $r(x)$ is convex in $[a_{(j)}; a_{(j+1)}]$ as a sum of m convex functions. Using well-known property of convex functions yields

$$\max_{x \in [a_{(j)}; a_{(j+1)}]} r(x) = \max\{r(a_{(j)}), r(a_{(j+1))}\}.$$

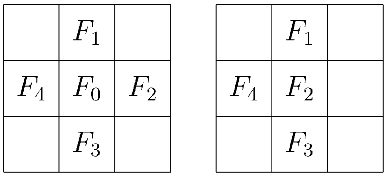
This finishes the proof. □

For the gray scale images we define the following fuzzy measure of similarity between two pixels F_k and F_l [10]:

$$\rho\{F_k, F_l\} = \mu(|F_k - F_l|). \tag{3}$$

Let us now assume that F_0 is the center pixel in the window W and that the pixels $F_1, F_2, \dots, F_{n-1}$ are surrounding F_0.

The filter works as follows. The central pixel F_0 is replaced by that F_{i_*} from the neighborhood of F_0 (Fig. 1), for which the total similarity function R_{i_*}



(a) (b)

Fig. 1. Illustration of the construction of the new filtering technique for the 4-neighborhood case. If the center pixel F_0 is replaced by its neighbor F_2, then the similarity measure $R_2 = \rho\{F_2, F_1\} + \rho\{F_2, F_3\} + \rho\{F_2, F_4\}$ between F_2 (new center pixel) is calculated. If the total similarity R_2 is greater than $R_0 = \rho\{F_0, F_1\} + \rho\{F_0, F_2\} + \rho\{F_0, F_3\} + \rho\{F_0, F_4\}$ then the center pixel F_0 is replaced by F_2, otherwise it is retained.

(which is a sum of all values of similarities between the central pixel and its neighbors) reaches its maximum. In other words if for some i

$$R_i = \sum_{j=1}^{n-1} (1 - \delta_{i,j})\rho(\mathbf{F}_i, \mathbf{F}_j), \quad i = 1, 2, \ldots, n-1, \tag{4}$$

is larger than

$$R_0 = \sum_{j=1}^{n-1} \rho(\mathbf{F}_0, \mathbf{F}_j), \tag{5}$$

then the center pixel is temporarily replaced by F_i. Generally, the pixel F_0 is given the value F_{i_*}, where $i_* = \arg\max R_i$

$$R_i = \delta_{i,0} \sum_{j=1}^{n-1} \rho(\mathbf{F}_i, \mathbf{F}_j) + (1 - \delta_{i,0}) \sum_{j=1}^{n-1} (1 - \delta_{i,j})\rho(\mathbf{F}_i, \mathbf{F}_j), \tag{6}$$

which means that the center pixel F_0 is replaced by that pixel from its neighborhood, for which the function R is being maximized.

This approach can be applied in a straightforward way to color images. We use the similarity function defined by $\rho\{\mathbf{F}_k, \mathbf{F}_l\} = \mu(||\mathbf{F}_k - \mathbf{F}_l)||$ where $||\cdot||$ is the specific vector norm. Now in exactly the same way we maximize the total similarity function R for the vector case.

In finding the maximum in Eq. (6), we obtain $n-1$ non-zero components in R_0. If we replace the central pixel by one of its neighbors (for instance by F_2 in Fig. 1a), then we obtain only $n-2$ non-zero components in R, as the pixel which has been put into the center disappears from the filter window (Fig. 1b). In this way the filter replaces the central pixel only when it is really noisy and preserves the original undistorted image structures.

3. Filter performance

The performance of the new algorithm was compared with the standard procedures of noise reduction used in color image processing. The color standard image *LENA* has been contaminated by 4% of impulsive noise. The impulsive noise has been simulated in two steps. In the first step each channel is corrupted independently with 4% impulsive noise. In the second step, a correlation factor $c = 0.5$ is used to further determine the corruption of a pixel (i, j) in a specific channel, if the same pixel (i, j) is corrupted in any of the two other channels. The second step simulates the channel correlation in multichannel images [8,11,12].

Table 1
Filters compared with the new noise reduction technique

Notation	Filter	Reference
AMF	Arithmetic mean filter	[2]
VMF	Vector median filter	[7]
ANNF	Adaptive nearest neighbor filter	[13]
BVDF	Basic vector directional filter	[8]
HDF	Hybrid directional filter	[9]
AHDF	Adaptive hybrid directional filter	[9]
DDF	Directional-distance filter	[5]
FVDF	Fuzzy vector directional filter	[14]

The root of the mean squared error ($RMSE$), normalized mean square error ($NMSE$) and peak signal-to-noise ratio ($PSNR$) have been used as quantitative measures of image quality for evaluation purposes.

We have checked several convex functions satisfying the conditions (1)–(3) in order to compare our approach with the standard filters used in color image processing presented in Table 1 and we have obtained the best results when applying the following similarity functions:

$$\mu_1(x) = \mathrm{e}^{-\beta_1 x}, \quad \beta_1 \in (0; \infty), \tag{7}$$

$$\mu_2(x) = \frac{1}{1 + \beta_2 x}, \quad \beta_2 \in (0; \infty), \tag{8}$$

$$\mu_3(x) = \frac{1}{(1 + x)^{\beta_3}}, \quad \beta_3 \in (0; \infty), \tag{9}$$

$$\mu_4(x) = -\frac{2}{\pi}\arctan(\beta_4 x) + 1, \quad \beta_4 \in (0; \infty), \tag{10}$$

$$\mu_5(x) = \frac{2}{1 + \mathrm{e}^{\beta_5 x}}, \quad \beta_5 \in (0; \infty), \tag{11}$$

$$\mu_6(x) = \frac{1}{1 + x^{\beta_6}}, \quad \beta_6 \in (0; 1), \tag{12}$$

$$\mu_7(x) = \begin{cases} 1 - \beta_7 x & \text{if } x < 1/\beta_7, \\ 0 & \text{if } x \geqslant 1/\beta_7, \end{cases} \quad \beta_7 \in (0; \infty). \tag{13}$$

Table 2 gives the optimal values of parameters β_i for the new filter. Table 3 summarizes the results obtained for the test image. We have used the L_2 norm and values of β_i from Table 2 and obtained the results shown

Table 2
Optimal values of constants β_i (10^{-3}) for the basic algorithm

β_1	β_2	β_3	β_4	β_5	β_6	β_7
5.04	6.62	192	6.97	7.90	266	3.72

Table 3
Comparison of the basic algorithm with the standard techniques (*LENA*)

Method	*NMSE* (10^{-4})	*RMSE*	*PSNR* (dB)
None	514.95	32.165	17.983
AMF	82.863	12.903	25.917
VMF	23.304	6.842	31.427
ANNF	31.271	7.926	30.149
BVDF	29.074	7.643	30.466
HDF	22.845	6.775	31.513
AHDF	22.603	6.739	31.559
DDF	24.003	6.944	31.288
FVDF	26.755	7.331	30.827
Proposed			
$\mu_1(x)$	**4.959**	**3.157**	**38.145**
$\mu_2(x)$	5.398	3.294	37.776
$\mu_3(x)$	9.574	4.387	35.288
$\mu_4(x)$	5.064	3.190	38.054
$\mu_5(x)$	**4.777**	**3.099**	**38.307**
$\mu_6(x)$	11.024	4.707	34.675
$\mu_7(x)$	**4.693**	**3.072**	**38.384**

Table 4
Comparison of the basic algorithm results (*RMSE*) using different norms (*LENA*)

	L_1	L_2	L_3	L_∞
$\beta_1(x)$	3.615	**3.157**	3.172	3.462
$\beta_5(x)$	3.579	**3.099**	3.167	3.694
$\beta_7(x)$	3.838	**3.072**	3.138	3.752

in Table 3. All proposed functions μ give very good results, although especially worth attention are μ_1, μ_5, μ_7. Table 4 shows the *RMSE* values obtained using the proposed filter for four different norms. As can be seen, the best choice is as usual L_2.

4. Adaptive selection of the parameter of the similarity function

The presented filter based on the similarity function is very effective in eliminating impulsive noise, almost as fast as the median filter and it is also very easy to implement.

It is clear that one of the possible ways to improve the filter's performance is to study the properties of the similarity function. We are now going to control the parameter of the similarity function in order to make the filtering more effective and dependent on the image structure and the fraction of corrupted image pixels.

The considerations presented below are established for the function

$$\mu_1(x) = e^{-\beta_1 x}, \quad \beta_1 > 0,$$

although the same method works for other μ_i as well.

The constant β_1 used to obtain results from Table 3 was chosen to obtain the best possible *PSNR* value for the *LENA* image with 4% impulsive noise. Although, as has been checked, the similarity function obtained in this way also enables good filtration of other images with different fractions of noise, it is natural to expect that the result might be improved when adapting the value of β_1 to the intensity of the noise process.

As it was already mentioned our filter has a good ability of preserving undamaged pixels. Table 5 shows the results of the filtering of the *LENA* image with 10% of pixels corrupted by random noise. The third row in the table contains fractions of image pixels which were changed by the filter. It is easy to see that the *PSNR* reaches its maximum for the value of β_1 for which the fraction of pixels changed by our filter is equal to the fraction of noisy pixels in the processed image. This property enables to adapt the value of β_1 for the best filter performance.

First, let us assume that the fraction p of noise in an image is known. In this case, the constant β_1 has to be set to the value, for which the percentage of pixels changed by the new filter is equal to p. In order to build a fast implementation of the adaptive filter version a well-known method of bisection [15] can be used.

This method allows finding of the root of an equation $g(x) = 0$ in $[a; b]$ providing that $g(x)$ is continuous and $g(a)g(b) < 0$.

In our case

$$g(\beta) = q(\beta) - p,$$

where $q(\beta)$ is the fraction of pixels changed by the filter.

The algorithm works as follows:

(1) Set $r := a$, $s := b$.
(2) Set $z := (r + s)/2$.
(3) If $g(z) = 0$ then output $\bar{\beta} = z$ and exit. In other case:
   (a) If $g(z)g(r) < 0$ then set $s := z$ and go to 2.
   (b) If $g(z)g(r) > 0$ then set $r := z$ and go to 2.

Obviously, the described process may be of infinite length and may not give an exact value, however it gives a good enough approximation of $\bar{\beta}$.

What is left, is to choose a starting interval $[a; b]$ and set the number of iterations of the algorithm, which provides us with the appropriate precision of $\bar{\beta}$.

Table 5
Noise reduction effect of the new filter (depending on parameter β_1) for the image *LENA* with 9.95% noisy pixels. The third row shows the fraction of pixels changed by filtration

β_1	10	1	0.1	0.01	0.006	0.005	0.004	0.001	0.0001
PSNR	28.95	29.03	30.00	37.03	38.63	38.76	38.51	23.81	18.64
%	74.34	73.32	44.31	10.60	9.97	9.83	9.66	4.47	0.00

Table 6
Comparison of the real and estimated fraction of noisy pixels

Real p	*LENA* Estimated p	*PEPPERS* Estimated p
0.01	0.0113	0.0122
0.02	0.0206	0.0216
0.05	0.0500	0.0510
0.10	0.0980	0.0986
0.20	0.1942	0.1964
0.40	0.3972	0.3973
0.70	0.7501	0.7504

For a wide range of the fractions of noisy pixels (from $p=0.01$ to more than 0.5) and for many standard color images, $g(0.001)<0$ and $g(0.01)>0$ holds, when a long enough interval is chosen: $a=0.001$, $b=0.01$.

The $\bar{\beta}$ has been taken after the fifth iteration of the bisection algorithm. Our experiments have shown that this yields good enough accuracy and leads to a significant improvement of the efficiency of the new filtering technique.

It is clear that the value of p is mostly not known, so we have to build an estimator for the percentage of the corrupted pixels p. The requirement for this estimator is that it must be able to find a value close to the fraction of changed pixels for a wide range of p (at least from $p=0.01$ to 0.50).

The following estimator is proposed. In the analysis of all the pixels which build an image, a pixel is considered to be undamaged by the noise process, if among eight of its neighbors, there exist at least two which are 'close' to it. Two pixels are said to be 'close' if the L_2 distance between them, in the RGB color space is less than 50 (we assume a true color, 24 Bit image).

As has been checked, this estimator works correctly. Table 6 shows the approximations of p given by the described estimator for two test color images (*LENA* and *PEPPERS*) and different fractions of corrupted pixels p. The full construction of the new filtering technique is as follows:

(1) Estimation of the fraction of corrupted pixels p.
(2) Finding an optimal value of $\bar{\beta}$ by using the method of bisection ($a=0,001$, $b=0,01$, five iterations).
(3) Filtration of an image using the similarity function, with the previously obtained parameter $\bar{\beta}$.

Table 7
Results of random noise reduction of the new filter compared with the vector median. The test image *LENA* was contaminated by 10%, 20%, 40% and 70% random noise (to x% of the pixels random RGB values were assigned)

	MAE	*RMSE*	*SNR*	*PSNR*
No filtering				
10%	7.714	29.814	13.542	18.642
20%	15.430	42.168	10.599	15.631
40%	31.113	59.865	7.625	12.587
70%	54.288	79.105	5.304	10.167
Vector median filter				
10%	3.496	6.086	27.343	32.444
20%	3.901	6.906	26.245	31.345
40%	5.135	8.965	23.979	29.080
70%	16.691	27.816	13.883	19.245
Proposed filter				
10%	0.587	3.351	35.523	37.627
20%	1.133	4.499	29.964	35.067
40%	2.560	7.093	26.003	31.114
70%	6.579	12.805	20.837	25.093

The effectiveness of this filter was tested using the standard *LENA* and *PEPPERS* images, with the percentage of damaged pixels ranging from 10% to 70%. The performance of the presented method was evaluated by means of the *MAE*, *RMSE*, *SNR* and *PSNR* coefficients. Table 7 depicts the obtained results.

The efficiency of the new filtering technique is shown in Figs. 2–7. Figs. 2 and 3 depict the results of image filtering using the new method in comparison with VMF. For the comparisons, the standard test images *LENA* and *PEPPERS* were used and the RGB channels were distorted by 4% impulsive noise.

Figs. 4 and 5 show the performance of the new filter using the same images distorted with 4% and 40% random noise (to x% of the image pixels random RGB values from the range $[0,255]$ were assigned).

Figs. 6 and 7 show the comparison of the new filtering method with the standard vector median filter using other color test images. As can be seen the filter is capable of reducing even strong random noise, while preserving the image details.

Fig. 2. Noise reduction effect of the proposed adaptive filter as compared with the standard VMF: (a) color test image *LENA*, (b) image distorted by 4% impulsive noise, (c) new adaptive method (3×3 window), (d) VMF, (e) and (f) the absolute difference between the original and filtered image (the RGB values were multiplied by factor 10).

5. Computational complexity of the new filter

Apart from the numerical behavior of any proposed algorithm, its computational complexity is a realistic measure of its practicality and usefulness, since it determines the required computing power and processing (execution) time.

A general framework to evaluate the computational requirements of image filtering algorithms is given in Refs. [16] and [17]. The framework of that analysis, originally introduced for filters utilizing a predefined moving window, is used here to evaluate the computational requirements of the algorithms.

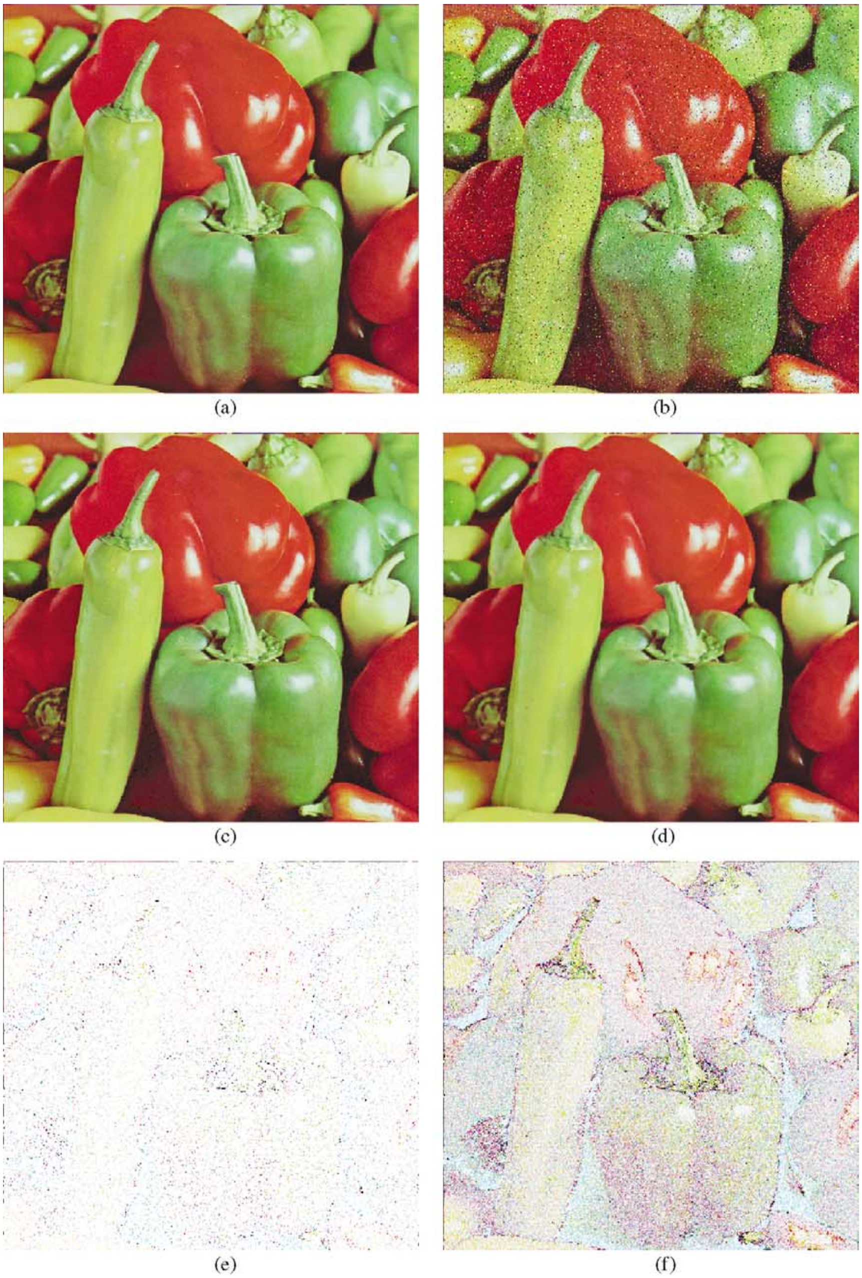


Fig. 3. Noise reduction effect of the proposed adaptive filter as compared with the standard VMF: (a) color test image *PEPPERS*, (b) image distorted by 4% impulsive noise, (c) new adaptive method (3×3 window), (d) VMF, (e) and (f) the absolute difference between the original and filtered image (the RGB values were multiplied by factor 10).

The requirement of this approach is that the filter window W is symmetric $(n \times n)$ and contains n^2 vector samples of order p $(\mathscr{R}^p)$. In most image processing applications a value $n = 3$ is considered.

The computational complexity of a specific filter is assumed to be a total time to complete an operation:

$$Time = \sum w_{OPER}\, OPER, \tag{14}$$

where *OPER* is the number of particular operations required and w_{OPER} is the weight of this operation.

In our analysis the following operations are used: *ADDS* (additions), *MULTS* (multiplications), *DIVS* (divisions), *SQRTS* (square roots), *COMPS* (comparisons), *ARCCOS* (arc cosines) and *EXPS* (exponents). The weights used in the calculations do not pertain to any particular machine. Rather, they can be considered

Fig. 4. Noise reduction effect of the proposed filter as compared with the VMF and DDF: (a) color test images, (b) images distorted by 4% random noise, (c) new method, (d) VMF, (e) DDF (3×3 window was used).

mean values of those coefficients commonly encountered. All qualitative results presented in the sequence hold, even if the weighting coefficients in the above formula are different for a specific computing platform. Mostly w_{ADDS} is assumed to be 1, while other w_{OPER} values depend on the computing platform and are out of our interest. In this way the computational complexity of the presented filter can be determined step-by-step as follows:

(1) Filtration of 1 pixel requires computation of n^2 total similarity measures $R(\mathbf{F}_j)$ and selection of their maximum ($n^2 - 1$ comparisons).
(2) Computation of one particular measure $R(\mathbf{F}_i)$ requires $n^2 - 2$ additions and $n^2 - 1$ calculations of $\rho\{\mathbf{F}_i, \mathbf{F}_j\}$.
(3) Computation of one particular $\rho\{\mathbf{F}_i, \mathbf{F}_j\}$ requires 1 computation of Euclidean distance (if the L_2 metric is used), 1 multiplication and 1 computation of an exponent.

Fig. 5. Noise reduction effect of the proposed filter as compared with the VMF and DDF: (a) color test images, (b) images distorted by 40% random noise, (c) new method, (d) VMF, (e) DDF (5 × 5 window was used).

(4) Computation of one particular Euclidean distance requires p multiplications, $2p$ additions and 1 square root.

Combining these steps, we conclude that the computational complexity of the presented method is

$$n^2((n^2-2)ADD + (n^2-1)(pMULT + 2pADD + SQRT + MULT + EXP)) + (n^2-1)COMP = O(n^4)MULT + O(n^4)ADD + O(n^4)SQRT) + O(n^4)EXP + O(n^2)COMP.$$

In the same way, we can obtain computational complexities for VMF and BVDF. Table 8 summarizes the results. As can be seen, the proposed filter has the same rank of complexity $O(n^4)$ as VMF and BVDF and is a little slower than VMF, but as fast as BVDF.

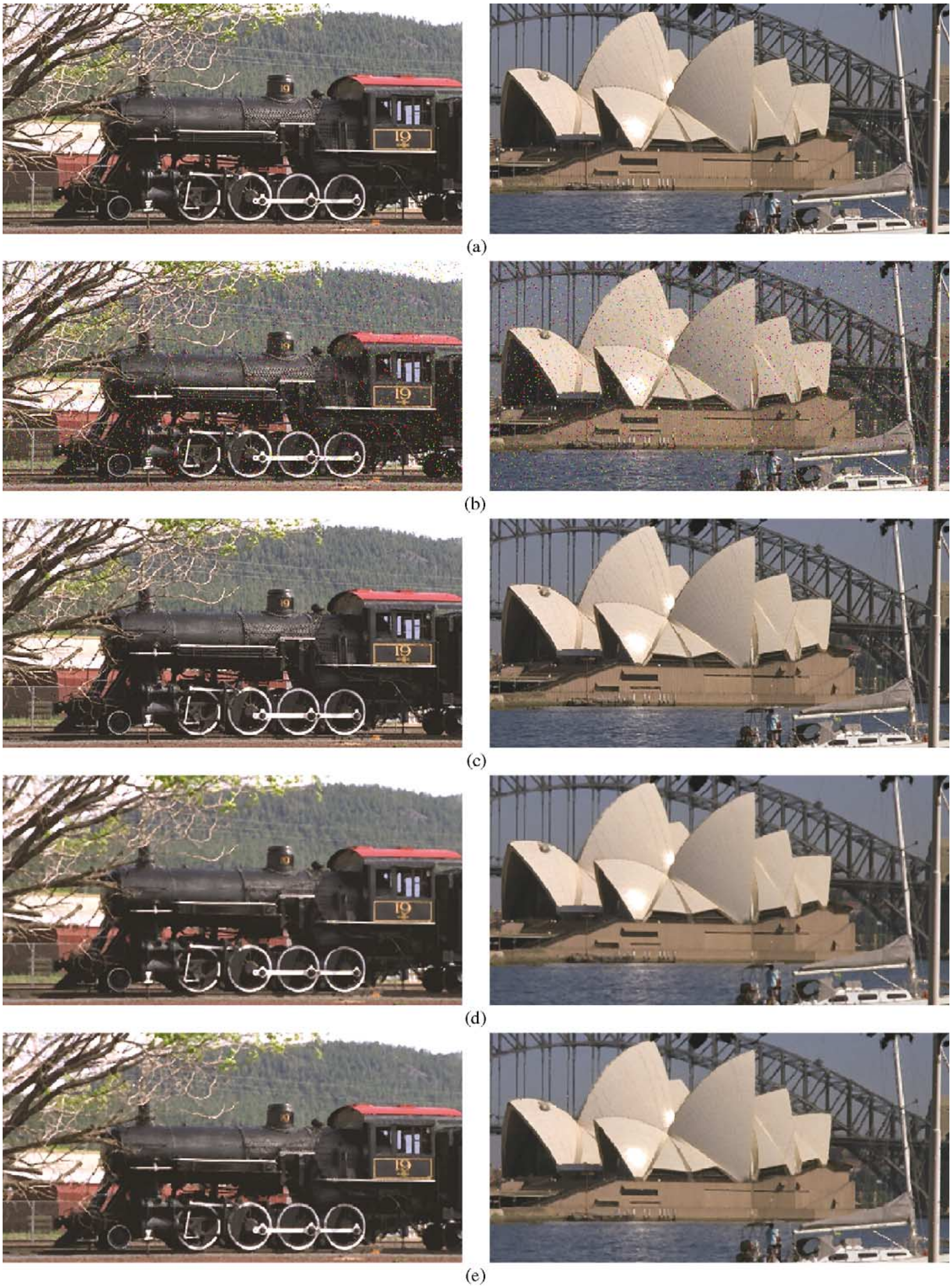
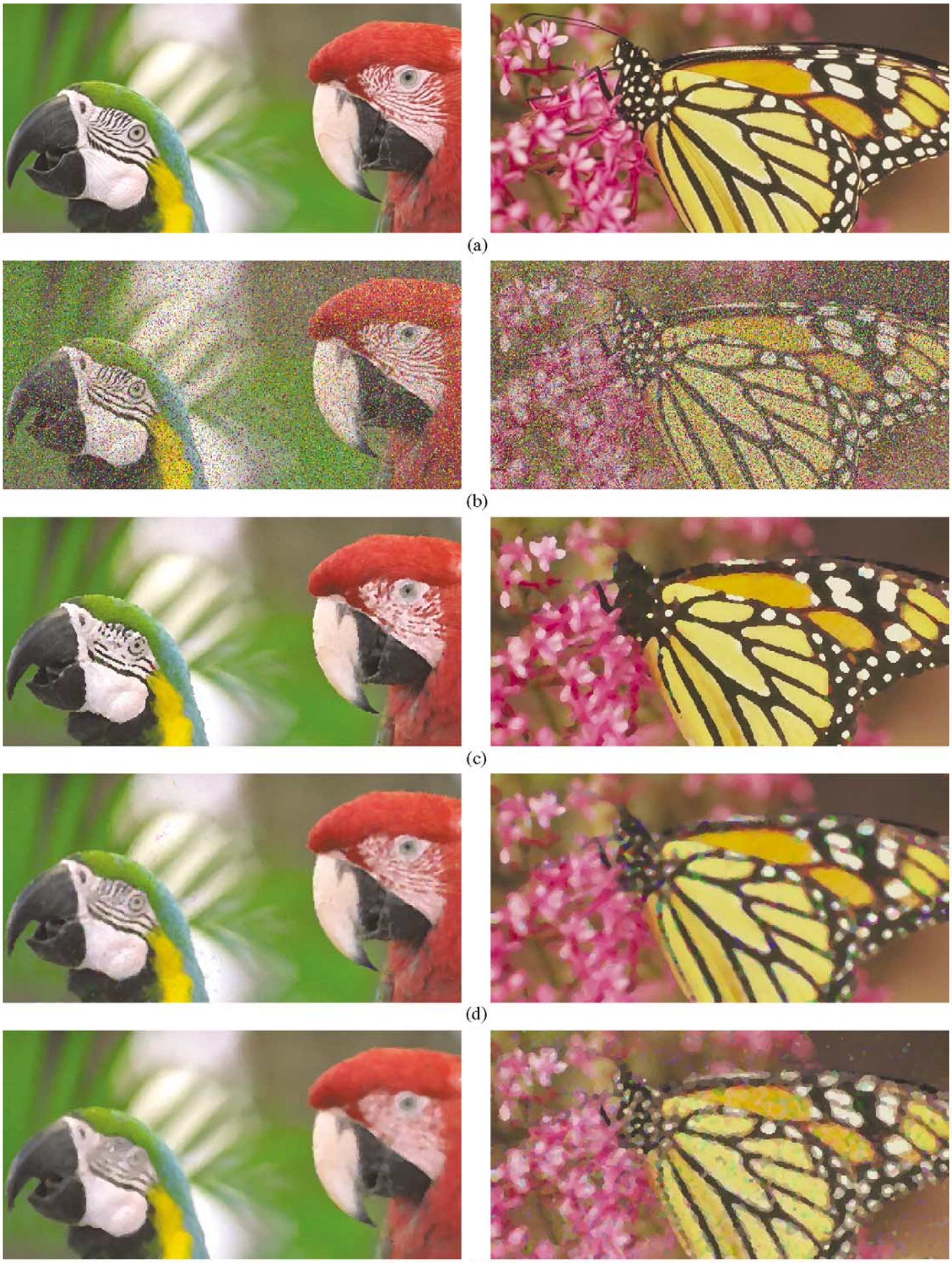


Fig. 6. Efficiency of the new filter: (a) color test images, (b) images distorted by 2% random noise, (c) new method, (d) VMF, (e) DDF (3×3 window was used).



(a)

(b)

(c)

(d)

(e)

Fig. 7. Noise reduction effect of the proposed filter as compared with the VMF: (a) color test images, (b) images distorted by 50% random noise, (c) new method (d) VMF, (e) DDF (5×5 window was used, 5 iterations were performed).

Table 8
Computational complexity of VMF, BVDF and proposed filter

	ADDS	*MULTS/DIVS*	*SQRTS*	*ARCCOS*	*EXPS*	*COMPS*
VMF	$O(n^4)$	—	—	—	—	$O(n^2)$
BVDF	$O(n^4)$	$O(n^4)$	$O(n^4)$	$O(n^4)$	—	$O(n^2)$
Proposed	$O(n^4)$	$O(n^4)$	$O(n^4)$	—	$O(n^4)$	$O(n^2)$

It must be stresses that all results were obtained by straightforward application of the described algorithms and are not optimal. For instance in Ref. [14], the way to reduce the complexity of VMF to $O(n^3)$ is described. Similar improvements might be applied to the presented filter as well.

The computational cost of the presented improvement of the filtering method based on the similarity approach, is significantly increased by the time required for the estimation of the percentage of the corrupted pixels, which allows to choose the optimal β value.

In applications in which the computational time is very important, the following solution is recommended. For finding of the optimal value of $\bar{\beta}$ using the method of bisection, not the whole image should be used, but only a small part of it. For instance a central square with 100×100 pixels. In this situation, the computational time needed for finding the optimal $\bar{\beta}$ is very small (in comparison with the final filtration) and then the method is almost as fast as the standard VMF. Generally, this simplification does not cause important changes to the effectiveness of the new filter.

6. Conclusions

The new algorithm presented in this paper can be seen as a modification and improvement of the commonly used vector median filter. The important advantage of this filter is connected with the following property: the algorithm adapts itself to the fraction of the impulsive noise in the image and to the structure of the image to achieve best possible performance.

The comparison shows that the new filter outperforms the basic standard procedures used in color image processing, when the impulse noise should be eliminated.

References

[1] A.N. Venetsanopoulos, K.N. Plataniotis, Multichannel image processing, Proceedings of the IEEE Workshop on Nonlinear Signal/Image Processing, 1995, pp. 2–6.

[2] I. Pitas, A.N. Venetsanopoulos, Nonlinear Digital Filters: Principles and Applications, Kluwer Academic Publishers, Boston, MA, 1990.

[3] K.N. Plataniotis, A.N. Venetsanopoulos, Color Image Processing and Applications, ISBN 3-540-66953-1, Springer, Berlin, 2000.

[4] I. Pitas, P. Tsakalides, Multivariate ordering in color image processing, IEEE Trans. Circuits Syst. Video Technol. 1, 3 (1991) 247–256.

[5] D.G. Karakos, P.E. Trahanias, Combining vector median and vector directional filters: The directional-distance filters, Proc. IEEE Conf. Image Process. ICIP-95 (1995) 171–174.

[6] I. Pitas, A.N. Venetsanopoulos, Order statistics in digital image processing, Proc. IEEE 80, 12 (1992) 1893–1923.

[7] J. Astola, P. Haavisto, Y. Neuovo, Vector median filters, IEEE Proc. 78 (1990) 678–689.

[8] P.E. Trahanias, A.N. Venetsanopoulos, Vector directional filters: A new class of multichannel image processing filters, IEEE Trans. Image Process. 2, 4 (1993) 528–534.

[9] M. Gabbouj, F.A. Cheickh, Vector median–vector directional hybrid filter for color image restoration, Proc. EUSIPCO-96 (1996) 879–881.

[10] B. Smolka, K. Wojciechowski, Random walk approach to image enhancement, Signal Process. 81 (3) (2001) 465–482.

[11] K.N. Plataniotis, D. Androutsos, A.N. Venetsanopoulos, Color image processing using adaptive vector directional filters, IEEE Trans. Circuits Systems-II: Analog Digital Signal Process. 45 (10) (1998) 1414–1419.

[12] K.N. Plataniotis, D. Androutsos, A.N. Venetsanopoulos, Fuzzy adaptive filters for multichannel image processing, Signal Process. J. 55 (1) (1996) 93–106.

[13] K.N. Plataniotis, D. Androutsos, V. Sri, A.N. Venetsanopoulos, A nearest neighbour multichannel filter, Electr. Lett. (1995) 1910–1911.

[14] K.N. Plataniotis, D. Androutsos, A.N. Venetsanopoulos, Colour image processing using fuzzy vector directional filters, Proceedings of the IEEE Workshop on Nonlinear Signal/Image Processing, Greece, 1995, pp. 535–538.

[15] D.M. Young, R.T. Gregory, A Survey of Numerical Mathematics, Vol. 1, Dover Publications, New York, 1988.

[16] M. Barni, V. Cappellini, On the computational complexity of multivariate median filters, Signal Process. 71 (1998) 45–54.

[17] K.N. Plataniotis, A.N. Venetsanopoulos, Vector processing. in: S.J. Sangwine, R.W.E. Horne (eds.), The Colour Image Processing Handbook, Chapman & Hall, Cambridge, Great Britain, 1997, pp. 188–209.

About the Author—BOGDAN SMOLKA received the Diploma in Physics degree from the Silesian University in Katowice, Poland in 1986 and Ph.D degree in Automatic Control from the Department of Automatic Control, Silesian University of Technology, Gliwice, Poland. From 1986 to 1989 he was a Teaching Assistant at the Department of Biophysics, Silesian Medical University, Katowice, Poland. From 1992 to 1994 he worked as a Teaching Assistant at the Technical University of Goeppingen, Germany. Since 1994 he has been with the Silesian University of Technology. In 1998 he was appointed as an Associate Professor at the Department of Automatic Control of the Silesian University of Technology. His current research interests include: application of Markov fields and random walks in low-level image processing, wavelet based methods of image compression and the visual aspects of image quality.

About the Author—KONSTANTINOS N. PLATANIOTIS received the B. Engineering degree in Computer Engineering from the Department of Computer Engineering and Informatics, University of Patras, Patras, Greece in 1988 and the M.S and Ph.D degrees in Electrical Engineering from the Florida Institute of Technology (Florida Tech), Melbourne, Florida in 1992 and 1994 respectively. He was a Research Associate with the Computer Technology Institute (C.T.I), Patras, Greece from 1989 to 1991 and a Postdoctoral Fellow at the Digital Signal & Image Processing Laboratory, Department of Electrical and Computer Engineering University of Toronto, from 1995 to 1997. From August 1997 to June 1999 he was an Assistant Professor with the School of Computer Science at Ryerson Polytechnic University. While at Ryerson, Prof. Plataniotis served as a lecturer in 12 courses to industry and Continuing Education programs. Since 1999 he has been with the University of Toronto as an Assistant Professor at the Department of Electrical & Computer Engineering where he researches and teaches adaptive systems and multimedia signal processing. He co-authored, with A.N. Venetsanopoulos, a book on “Color Image Processing & Applications”, Springer Verlag, August 2000, ISBN 3-540-66953-1, he is a contributor to three books, and he has published more than 100 papers in refereed journals and conference proceedings on the areas of adaptive systems, signal and image processing, and communication systems and stochastic estimation. Prof. Plataniotis is a member of the IEEE Technical Committee on Neural Networks for Signal processing, and the Technical Co-Chair of the Canadian Conference on Electrical and Computer Engineering, CCECE 2001, May 13–16, 2001, Toronto, Ontario.

His current research interests include: adaptive systems statistical pattern recognition, multimedia data processing, statistical communication systems, and stochastic estimation and control.

About the Author—ANDRZEJ CHYDZINSKI received the Diploma in Engineering degree and M.S. in Applied Mathematics from the Silesian University of Technology, Gliwice, Poland in 1997 and is a Ph.D. student at the moment. Since 1997, he has been with the Silesian University of Technology at the Department of Mathematics, where he researches and teaches probability theory and its applications in computer sciences; specially in image processing, data compression and queueing systems. He co-authored, with M. Bratiychuk, a book on “Probability Theory”, (edition in December 2000) and he has published several papers in refereed journals and conference proceedings on the mentioned subjects.

About the Author—MAREK SZCZEPANSKI received the Diploma in Engineering Degree and M.S. in Automatic Control and Robotics from the Silesian University of Technology, Gliwice, Poland in 1998 and is a Ph.D. student at the moment. Since 1998 he has been with the Silesian University of Technology at the Department of Automatic Control, where he is involved in image processing and pattern recognition, control theory and artificial intelligence research. His current research interests include: application of Markov fields and random walks in low-level image processing, multichannel image processing, image quality and biomedical image processing.

About the Author—KONRAD WOJCIECHOWSKI received the M.Sc. Diploma in Electrical Engineering from the Academy of Mining and Metallurgy, Cracow, Poland in 1967 and the Ph.D., D.Sc. (habilitation dissertation) in Control Theory from Silesian University of Technology, Gliwice, Poland in 1976 and 1991, respectively. He joined the Department of Automatic Control, Electronics and Computer Science, Silesian University of Technology in 1968 as a doctoral student and he was promoted to Assistant Professor in 1976, and Professor in 1991. He serves lectures in linear and nonlinear control theory, adaptive control, image processing and pattern recognition, signal processing and elements of artificial intelligence. He has been awarded by the Chancellor for scientific results and received an Award of the Polish Ministry of Education. Since 1995 he joined Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Gliwice. He has published 150 papers in refereed journals and conference proceedings on control theory, decision making, image processing and digital communications. In 1996, Prof. K. Wojciechowski was elected as the Dean of the Department of Automatic Control, Electronics and Computer Science. Prof. K. Wojciechowski is the head of Laboratory of Computer Vision, Chair of doctoral studies in “Automation and Robotics” and Chair of postgraduate studies in “Spatial Information Systems”. He has been director of Summer School on Morphological Image and Signal Processing and member of Programme Committees and Organising Committees of numerous conferences. He is a member of the Execution Board of the Society of Image Processing and member of Polish Electrical Engineering Society. His current research interests include: control and adaptive systems, control based on visual information, artificial intelligence, image processing, spatial information systems and multimedia.

About the Author—ANASTASIOS N. VENETSANOPOULOS received the Diploma in Engineering degree from the National Technical University of Athens (NTU), Greece, in 1965, and the M.S., M.Phil., and Ph.D. degrees in Electrical Engineering from Yale University in 1966, 1968 and 1969, respectively. He joined the Department of Electrical and Computer Engineering of the University of Toronto in September 1968 as a Lecturer and he was promoted to Assistant Professor in 1970, Associate Professor

in 1973, and Professor in 1981. Since July 1997, he has been Associate Chair: Graduate Studies of the Edward S. Rogers Sr. Department of Electrical and Computer Engineering and was Acting Chair during the spring term of 1998–99. In 1999 a Chair in Multimedia was established in the ECE Department, made possible by a donation from Bell Canada, and Prof. A.N. Venetsanopoulos assumed the position as Inaugural Chairholder in July 1999. He has served as Chair of the Communications Group and Associate Chair of the Department of Electrical Engineering and Associate Chair: Graduate Studies for the Department of Electrical and Computer Engineering. Prof. A.N. Venetsanopoulos was on research leave at Imperial College of Science and Technology, the National Technical University of Athens, the Swiss Federal Institute of Technology, the University of Florence and the Federal University of Rio de Janeiro, and has also served as Adjunct Professor at Concordia University. He has served as lecturer in 138 short courses to industry and continuing education programs and as Consultant to numerous organizations; he is a contributor to twenty eight (28) books, a co-author of *Color Image Processing and Applications*, ISBN 3-540-66953-1, Springer Verlag, August 2000, *Nonlinear Filters in Image Processing*: *Principles Applications*, ISBN-0-7923-9049-0, Kluwer, 1990, *Artificial Neural Networks*: *Learning Algorithms*, *Performance Evaluation and Applications*, ISBN-0-7293-9297-3, Kluwer, 1993, and *Fuzzy Reasoning in Information Decision and Control systems*, ISBN-0-7293-2643-1, Kluwer 1994. He has published 750 papers in refereed journals and conference proceedings on digital signal and image processing, and digital communications. Prof. Venetsanopoulos has served as Chair on numerous boards, councils and technical conference committees of the Institute of Electrical and Electronic Engineers (IEEE), such as the Toronto Section (1977–1979) and the IEEE Central Canada Council (1980–1982); he was President of the Canadian Society for Electrical Engineering and Vice President of the Engineering Institute of Canada (EIC) (1983–1986). He was a Guest Editor or Associate Editor for several IEEE journals and the Editor of the Canadian Electrical Engineering Journal (1981–1983). Prof. A.N. Venetsanopoulos is the Technical Program Co-Chair of the IEEE International Conference on Image Processing, ICIP-2001, to be held in Thessaloniki, Greece, October 7–10, 2001. He is a member of the IEEE Communications, Circuits and Systems, Computer, and Signal Processing Societies of IEEE, as well as a member of Sigma Xi, the Technical Chamber of Greece, the European Association of Signal Processing, the Association of Professional Engineers of Ontario (APEO) and Greece. He was elected as a Fellow of the IEEE “for contributions to digital signal and image processing”, he is also a Fellow of the EIC, “for contributions to electrical engineering”, and was awarded an Honorary Doctorate from the National Technical University of Athens, in October 1994. In October 1996, he was awarded the “Excellence in Innovation Award” of the Information Technology Research Centre of Ontario and Royal Bank of Canada, “for innovative work in color image processing”.